
APPENDIX D

Install Python 3

By the time Python 3 is preinstalled on every machine, toasters will be replaced by 3-D printers that crank out daily doughnuts with sprinkles. Windows doesn't have Python at all, and OS X, Linux, and Unix tend to have old versions. Until they catch up, you'll probably need to install Python 3 yourself.

The following sections describe how to carry out these tasks:

- Find out what version of Python you have on your computer, if any
- Install the standard distribution of Python 3, if you don't have it
- Install the Anaconda distribution of scientific Python modules
- Install `pip` and `virtualenv`, if you can't modify your system
- Install `conda` as an alternative to `pip`

Most of the examples in this book were written and tested with Python 3.3, the most recent stable version at the time of writing. Some used 3.4, which was released during the editing process. The [What's New in Python page](#) presents what was added in each version. There are many sources of Python and many ways to install a new version. In this appendix, I'll describe two of these ways:

- If you just want the standard interpreter and libraries, I recommend going to the [official language site](#).
- If you would like Python together with the standard library, and the great scientific libraries described in [Appendix C](#), use Anaconda.

Install Standard Python

Go to the Python [download page](#) with your web browser. It tries to guess your operating system and present the appropriate choices, but if it guesses wrong, you can use these:

- [Python Releases for Windows](#)
- [Python Releases for Mac OS X](#)
- [Python Source Releases \(Linux and Unix\)](#)

You'll see a page similar to that shown in [Figure D-1](#).

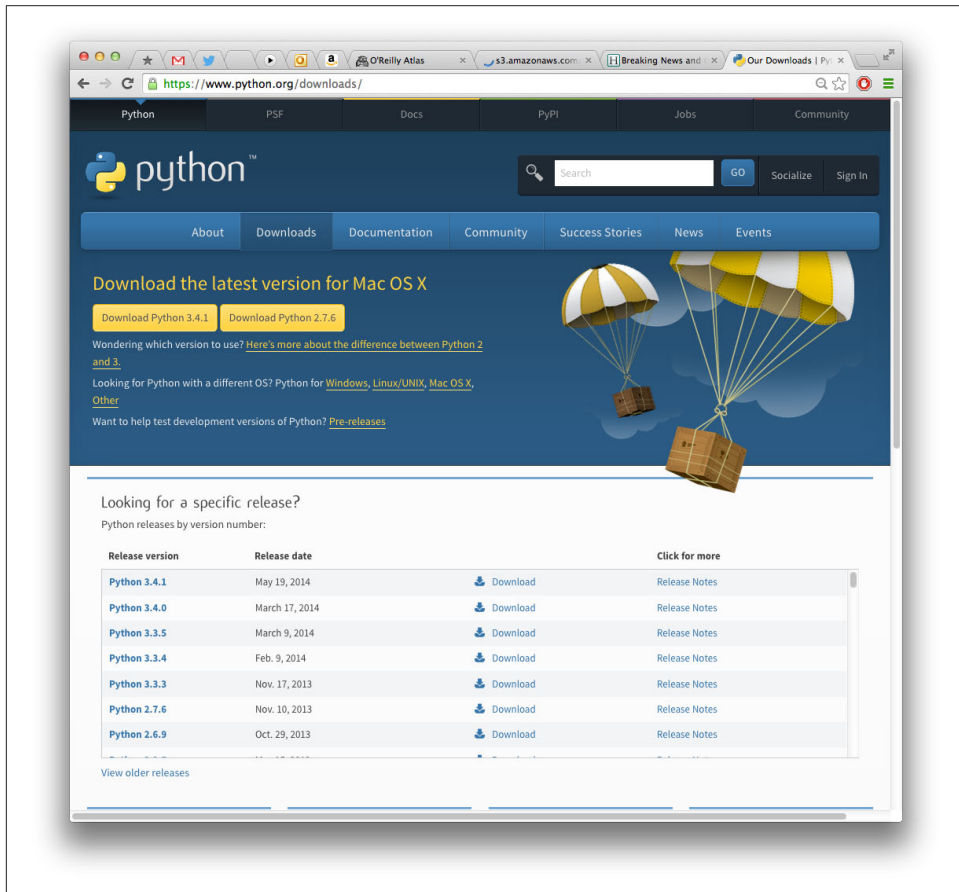


Figure D-1. Sample download page

Click the Download link for the most recent version. In our case, that's 3.4.1. This takes you to an information page like the one shown in [Figure D-2](#).



Figure D-2. Detail page for download

You need to scroll down the page to see the actual download link (Figure D-3).

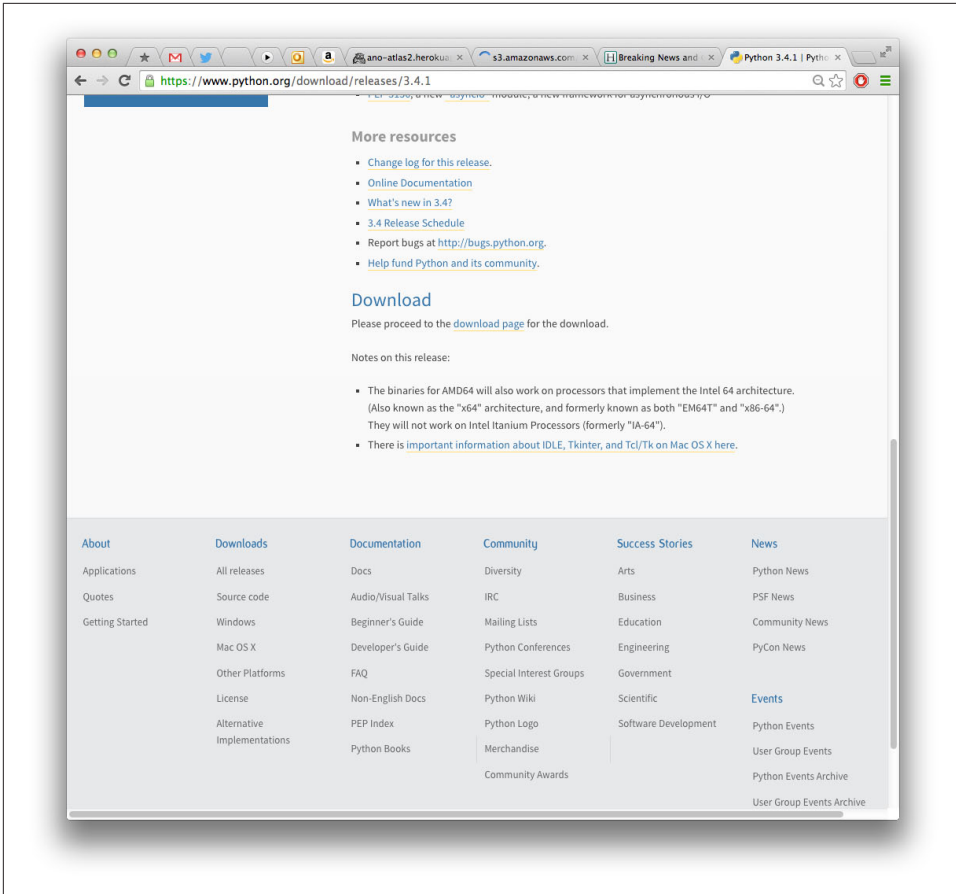


Figure D-3. Bottom of page offering download

Click the download link to get to the actual release-specific page (Figure D-4).

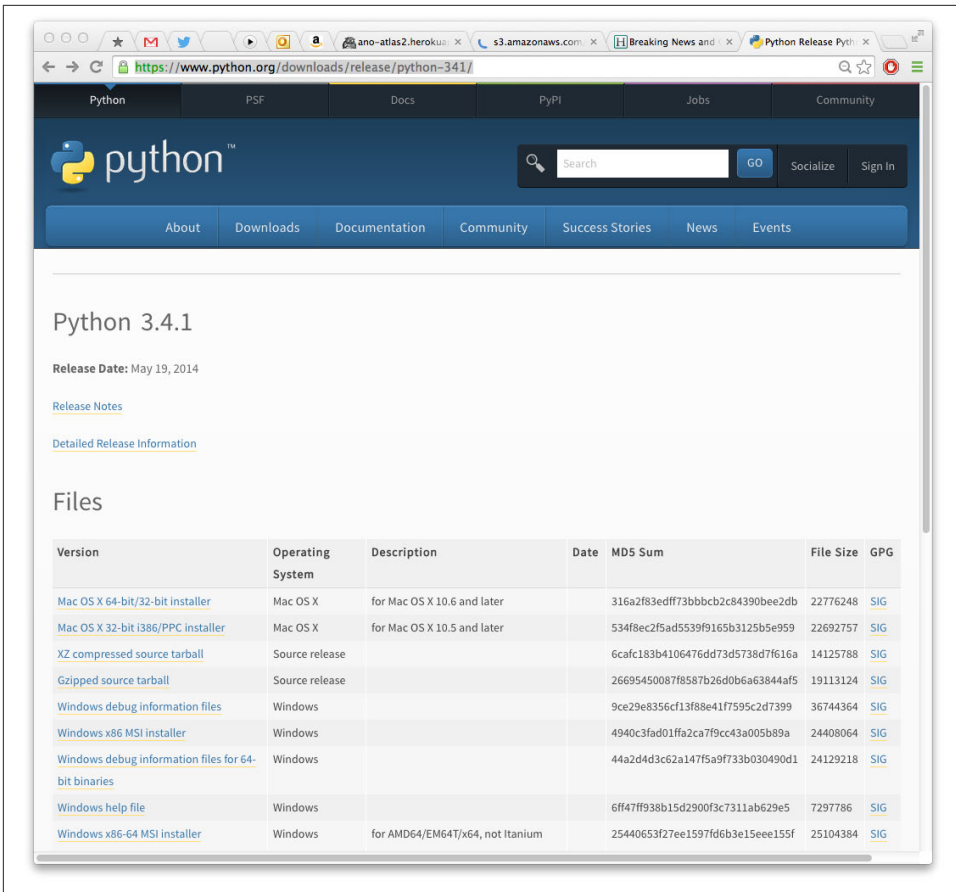


Figure D-4. Files to download

Now, click the correct version for your computer.

Mac OS X

Click the **Mac OS X 64-bit/32-bit installer** link to download a Mac *.dmg* file. Double-click it after the download completes. A window with four icons opens. Right-click *Python.mpkg* and then, in the dialog box that opens, click Open. Click the Continue button three times or so to step through the legalese, and then, when the appropriate dialog box opens, click Install. Python 3 will be installed as `/usr/local/bin/python3`, leaving any existing Python 2 on your computer unchanged.

Windows

For Windows, download one of these:

- [Windows x86 MSI installer \(32-bit\)](#)
- [Windows x86-64 MSI installer \(64-bit\)](#)

To determine whether you have a 32-bit or 64-bit version of Windows, do the following:

1. Click the Start button.
2. Right-click Computer.
3. Click Properties and find the bit value.

Click the appropriate installer (.msi file). After it's downloaded, double-click it and follow the installer directions.

Linux or Unix

Linux and Unix users get a choice of compressed source formats:

- [XZ compressed source tarball](#)
- [Gzipped source tarball](#)

Download either one. Decompress it by using `tar -xJ (.xz file)` or `tar -xz (.tgz file)` and then run the resulting shell script.

Install Anaconda

Anaconda is an all-in-one installer with an emphasis on science: it includes Python, the standard library, and many useful third-party libraries. Until recently, it included Python 2 as its standard interpreter, although there was a workaround to install Python 3.

The new upgrade, Anaconda 2.0, installs the latest version of Python and its standard library (3.4 when this was written). Other goodies include libraries that we've talked about in this book: `beautifulsoup4`, `flask`, `ipython`, `matplotlib`, `nose`, `numpy`, `pandas`, `pillow`, `pip`, `scipy`, `tables`, `zmq`, and many others. It includes a cross-platform installation program called `conda` that improves on `pip`; we'll talk about that shortly.

To install Anaconda 2, go to the [download page](#) for the Python 3 versions. Click the appropriate link for your platform (version numbers might have changed since this was written, but you can figure it out):

- To download for the Mac, click [Anaconda3-2.0.0-MacOSX-x86_64.pkg](#). Double-click the file when it's done downloading, and then follow the usual steps for installing Mac software. It will install everything to the *anaconda* directory under your home directory.

- For Windows, click the **32-bit version** or **64-bit version**. Double-click the `.exe` file after it downloads.
- For Linux, click the **32-bit version** or **64-bit version**. When it has downloaded, execute it (it's a big shell script).



Ensure that the name of the file you download starts with *Anaconda3*. If it starts with just *Anaconda*, that's the Python 2 version.

Anaconda installs everything to its own directory (*anaconda* under your home directory). This means that it won't interfere with any versions of Python that might already be on your computer. It also means that you don't need any special permission (names like `admin` or `root`) to install it either.

To see what packages are included, visit [the Anaconda docs page](#) and then, in the box at the top of the page, click "Python version: 3.4." It listed 141 packages when I last looked.

After installing Anaconda 2, you can see what Santa put on your computer by typing this command:

```
$ ./conda list
# packages in environment at /Users/williamlubanovic/anaconda:
#
anaconda                2.0.0                np18py34_0
argcomplete             0.6.7                py34_0
astropy                 0.3.2                np18py34_0
backports.ssl-match-hostname 3.4.0.2             <pip>
beautiful-soup          4.3.1                py34_0
beautifulsoup4          4.3.1                <pip>
binstar                 0.5.3                py34_0
bitarray                0.8.1                py34_0
blaze                   0.5.0                np18py34_0
blz                     0.6.2                np18py34_0
bokeh                   0.4.4                np18py34_1
cdecimal                2.3                  py34_0
colorama                 0.2.7                py34_0
conda                   3.5.2                py34_0
conda-build             1.3.3                py34_0
configobj               5.0.5                py34_0
curl                    7.30.0               2
cython                  0.20.1               py34_0
datashape               0.2.0                np18py34_1
dateutil                2.1                  py34_2
docutils                0.11                 py34_0
```

dynd-python	0.6.2	np18py34_0
flask	0.10.1	py34_1
freetype	2.4.10	1
future	0.12.1	py34_0
greenlet	0.4.2	py34_0
h5py	2.3.0	np18py34_0
hdf5	1.8.9	2
ipython	2.1.0	py34_0
ipython-notebook	2.1.0	py34_0
ipython-qtconsole	2.1.0	py34_0
itsdangerous	0.24	py34_0
jdcal	1.0	py34_0
jinja2	2.7.2	py34_0
jpeg	8d	1
libdynd	0.6.2	0
libpng	1.5.13	1
libsodium	0.4.5	0
libtiff	4.0.2	0
libxml2	2.9.0	1
libxslt	1.1.28	2
llvm	3.3	0
llvmpy	0.12.4	py34_0
lxml	3.3.5	py34_0
markupsafe	0.18	py34_0
matplotlib	1.3.1	np18py34_1
mock	1.0.1	py34_0
multipledispatch	0.4.3	py34_0
networkx	1.8.1	py34_0
nose	1.3.3	py34_0
numba	0.13.1	np18py34_0
numexpr	2.3.1	np18py34_0
numpy	1.8.1	py34_0
openpyxl	2.0.2	py34_0
openssl	1.0.1g	0
pandas	0.13.1	np18py34_0
patsy	0.2.1	np18py34_0
pillow	2.4.0	py34_0
pip	1.5.6	py34_0
ply	3.4	py34_0
psutil	2.1.1	py34_0
py	1.4.20	py34_0
pycosat	0.6.1	py34_0
pycparser	2.10	py34_0
pycrypto	2.6.1	py34_0
pyflakes	0.8.1	py34_0
pygments	1.6	py34_0
pyparsing	2.0.1	py34_0
pyqt	4.10.4	py34_0
pytables	3.1.1	np18py34_0
pytest	2.5.2	py34_0
python	3.4.1	0
python-dateutil	2.1	<pip>

python.app	1.2	py34_2
pytz	2014.3	py34_0
pyyaml	3.11	py34_0
pyzmq	14.3.0	py34_0
qt	4.8.5	3
readline	6.2	2
redis	2.6.9	0
redis-py	2.9.1	py34_0
requests	2.3.0	py34_0
rope	0.9.4	py34_1
rope-py3k	0.9.4	<pip>
runipy	0.1.0	py34_0
scikit-image	0.9.3	np18py34_0
scipy	0.14.0	np18py34_0
setuptools	3.6	py34_0
sip	4.15.5	py34_0
six	1.6.1	py34_0
sphinx	1.2.2	py34_0
spyder	2.3.0rc1	py34_0
spyder-app	2.3.0rc1	py34_0
sqlalchemy	0.9.4	py34_0
sqlite	3.8.4.1	0
ssl_match_hostname	3.4.0.2	py34_0
sympy	0.7.5	py34_0
tables	3.1.1	<pip>
tk	8.5.15	0
tornado	3.2.1	py34_0
ujson	1.33	py34_0
werkzeug	0.9.4	py34_0
xlrd	0.9.3	py34_0
xlsxwriter	0.5.5	py34_0
yaml	0.1.4	1
zeromq	4.0.4	0
zlib	1.2.7	1

Install and Use pip and virtualenv

The `pip` package is the most popular way to install third-party (nonstandard) Python packages. It has been annoying that such a useful tool isn't part of standard Python, and that you've needed to download and install it yourself. As a friend of mine used to say, it's a cruel hazing ritual. The good news is that `pip` is a standard part of Python, starting with the 3.4 release.

Often used with `pip`, the `virtualenv` program is a way to install Python packages in a specified directory (folder) to avoid interactions with any preexisting system Python packages. This lets you use whatever Python goodies you want, even if you don't have permission to change the existing installation.

If you have Python 3 but only the Python 2 version of `pip`, here's how to get the Python 3 version on Linux or OS X:

```
$ curl -O http://python-distribute.org/distribute_setup.py
$ sudo python3 distribute_setup.py
$ curl -O https://raw.githubusercontent.com/pypa/pip/master/contrib/get-pip.py
$ sudo python3 get-pip.py
```

This installs pip-3.3 in the bin directory of your Python 3 installation. Then, use pip-3.3 to install third-party Python packages rather than Python 2's pip.

Some good guides to pip and virtualenv are:

- [A non-magical introduction to Pip and Virtualenv for Python beginners](#)
- [The hitchhiker's guide to packaging: pip](#)

Install and Use conda

Until recently, pip always downloaded source files rather than binaries. This can be a problem with Python modules that are built on C libraries. Recently, the Anaconda developers built **conda** to address the problems they've seen with pip and other tools. pip is a Python package manager, but conda works with any software and language. conda also avoids the need for something like virtualenv to keep installations from stepping on one another.

If you installed the Anaconda distribution, you already have the conda program. If not, you can get Python 3 and conda from the [miniconda page](#). As with Anaconda, make sure the file you download starts with MiniConda3; if it starts with MiniConda alone, it's the Python 2 version.

conda works with pip. Although it has its own public [package repository](#), commands like `conda search` will also search the [PyPi repository](#). If you have problems with pip, conda might be a good alternative.